

18.6 Data Recovery and Retiming for the Fully Buffered DIMM 4.8Gb/s Serial Links

Hamid Partovi¹, Wolfgang Walthes², Luca Ravezzi¹, Paul Lindt², Sivaraman Chokkalingam¹, Karthik Gopalakrishnan¹, Andreas Blum², Otto Schumacher², Claudio Andreotti², Michael Bruennert², Bruno Celli-Urbani², Dirk Friebe², Ivo Koren², Michael Verbeck², Ulrich Lange²

¹Infineon Technologies, San Jose, CA

²Infineon Technologies, Munich, Germany

The increasing demand for DRAM capacity and performance in computing, and especially servers, has led to the development of a new memory-interface standard, the fully buffered DIMM (FB-DIMM).

FB-DIMMs can host up to 36 DRAMs whose communication to the host processor is facilitated by the advanced memory buffer (AMB). While DRAMs on a DIMM interact with their respective AMB using the conventional DDR2 standard, the AMB sends to and receives data from the host processor or a neighboring FB-DIMM by means of differential point-to-point signaling.

In this paper, the implementation details of data recovery and retiming of the AMB serial links are discussed. The chip comprises 24 serial links, a core processing unit, and a DDR interface. To support an 800Mb/s DDR2 data rate, links must operate at 4.8Gb/s. FB-DIMMs are connected in a daisy-chain configuration, and as such, the serial links function as repeaters; they recover and retime data, process, and forward data to the next DIMM, starting from and ending at the host processor. Figure 18.6.1 depicts the block diagram of a single high-speed lane including the CDR, electrical idle, the IQ-generator, a retiming FIFO, and the transmitter.

The FB-DIMM protocol uses electrical idle (EI) as the primary mechanism to initialize, control state transitions, and to enter and exit the disable state. AMB enters EI when both the differential (DM) and the common-mode (CM) levels of the received data on at least two of three assigned links are low.

The key challenge with the EI-detection circuit is its required resolution and bandwidth. The EI must detect the valid, but deteriorated differential levels ($\pm 80\text{mV}$) of serial data in the presence of considerable CM noise both in EI and active modes; and with fast response time, it must determine whether the incoming data stream is valid or if the preceding AMB is in idle state.

Figure 18.6.2 is a simplified schematic of the EI circuit illustrating only the *differential* level detection. CMFB biases the gates of drain-coupled devices, Md+ and Md- near V_i when DM=0. With the application of a differential data stream, Md+ and Md- gates alternate above V_i . Acting as a wideband full-wave rectifier, the pair generates a current, I_{int}, which is in turn dc-averaged by the RC load to effect a voltage drop on V_{int}. Replica biasing produces V_{intR} to which V_{int} is compared in order to indicate entry into or exit from EI. As seen from the figure, though the input instantaneous voltage level in the active mode is frequently below that of EI, the circuit never makes a false transition, and achieves entry and exit detection times of 16ns and 8ns over PVT and mismatch, outperforming the specification of 60ns and 30ns [1].

A half-rate (2.4GHz) CML clock is distributed to pairs of lanes and is used to generate, by means of a polyphase filter (Fig. 18.6.3), quadrature clocks that drive two adjacent phase interpolators (PIs). Worst case IQ error is 0.015UI, or 3ps, and duty-cycle error is less than 0.5%. Phase interpolation is achieved by quadrant-based phase-mixing with a resolution of 1/32 UI and a DNL better than 0.25 LSB with a $\pm 3\sigma$ confidence level over PVT. The half-rate CML clock is also converted to CMOS levels for use by the high-speed digital circuits, and the transmitter.

Much like the EI detection time, fast acquisition of lock that ensues on exit from EI, significantly improves system performance after reset and recovery.

The AMB uses a 1st-order tracking CDR with fast acquisition capability based on *binary search* (see Fig. 18.6.4). The algorithm is independent of the loop delay, and thus enables a very short acquisition time without exhibiting any limit-cycle oscillations. Excepting the digital loop filter that operates at the decimated frequency of 600MHz, and comprises low-bandwidth tracking and fast acquisition modules, this architecture requires no additional high-speed components when compared to a generic CDR, and thus affords considerable savings in area and power.

While the tracking filter receives the *difference* of Up<7:0> and Dn<7:0> counts, the fast acquisition module *separately* integrates the Up<7:0> and Dn<7:0> counts using a pair of shallow dumped integrators (DIs).

The lock condition is reached with three successive steps; in each step, the first DI to cross the threshold indicates in which direction the recovered clock phase must be shifted. Adjustments are executed by trains of 8, 4, and 2 Up_{Acq} or Dn_{Acq} pulses sent to the PI, which, in turn, shifts the recovered clock by $\pm 1/4\text{UI}$ ($\pm 8\text{LSBs}$), $\pm 1/8\text{UI}$ ($\pm 4\text{LSBs}$), and $\pm 1/16\text{UI}$ ($\pm 2\text{LSBs}$). Though non-monotonic, the residual phase error is always less than the respective correction step, and is reduced to within 2LSBs at the end of the acquisition process.

During each step, for the time the PI adjusts the clock phase, the loop is *broken* (i.e., both DIs are cleared and are held in reset), and is reconnected once the interpolator has settled. Such procedure eliminates any possibility of limit-cycle oscillations in the CDR behavior.

Upon completion of the 3rd step, the FSM asserts the "Lock Detect" signal and enables the low-bandwidth tracking loop which will complete the final phase convergence. Contemporaneously, the FIFO and transmitter are enabled. Figure 18.6.5 includes the phase convergence process during fast acquisition for the full span of initial phase errors [-1/2UI, +1/2UI]. The fast lock process completes in 520UIs, well exceeding the standards requirement of 1428UIs [2].

A Retiming FIFO receives the recovered clock and data. It interrupts the accumulation of jitter in the FB-DIMM daisy-chain by retiming the recovered data to the local PLL clock. Optionally, the AMB can bypass the FIFO, and forward the recovered data without retiming to the transmitter.

As thru-latency is one of the key performance parameters of AMB, the FIFO is designed to operate at 2.4GHz with both writes and reads accomplished in half a period (1UI). The FIFO (Fig. 18.6.6) is implemented as a 2-entry, 8-deep, dual-port register-file. By integrating an insertion MUX onto its read bit lines, data from the DDR interface can selectively be inserted and forwarded to the link transmitter. Read and write operations are differential and utilize monotonic, dual-rail domino signaling.

A pair of ring counters generates the write and read pointers. Write-to-Read pointer spacing is programmable to 2, 3 or 4UIs so that the lowest setting, based on the expected accumulated jitter, can be selected.

The AMB die, shown in Fig. 18.6.7, is fabricated in a 0.13 μm , 1.5V CMOS technology and occupies $9.2 \times 4.5\text{mm}^2$. The measured input sensitivity, with a minimum eye-opening of 0.35UI, is 50mV_{p-p} at a BER of 10⁻¹², and is better than 170mV_{p-p} for an extrapolated BER of 10⁻¹⁶, exceeding the standards requirement of 170mV_{p-p} at a BER of 10⁻¹² [1]. Limited only by the available chipset support, a cascade of up to 4 FB-DIMMs interoperates with a host processor without error.

References:

- [1] FB-DIMM Draft Specification: *High Speed Differential PTP Link at 1.5V*, Dec., 04.
- [2] FB-DIMM Draft Specification: *AMB*, Jan., 05.

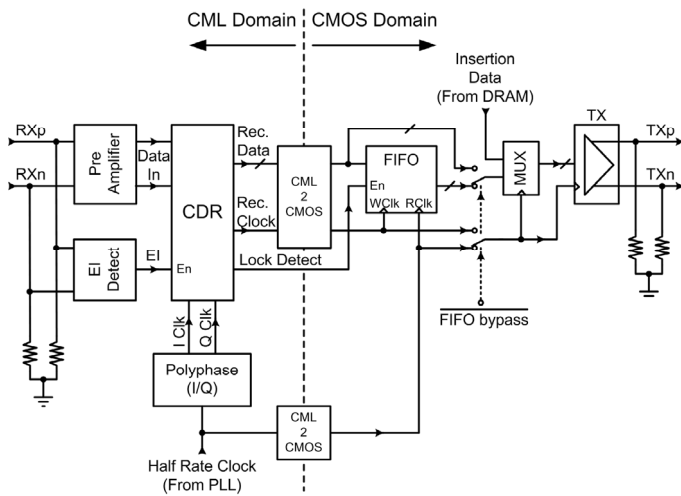


Figure 18.6.1: High-speed lane architecture.

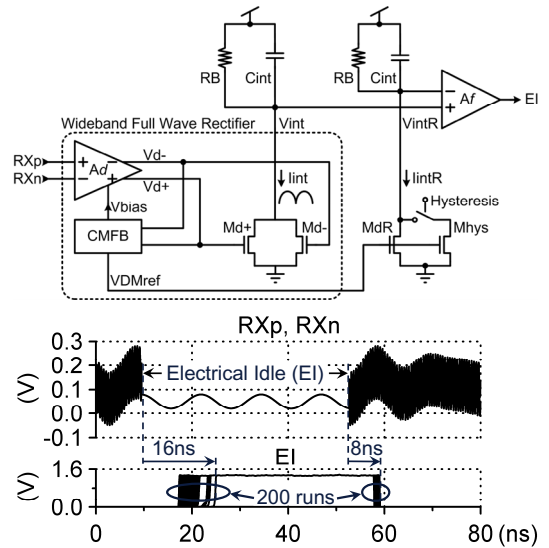


Figure 18.6.2: Electrical idle detection circuit.

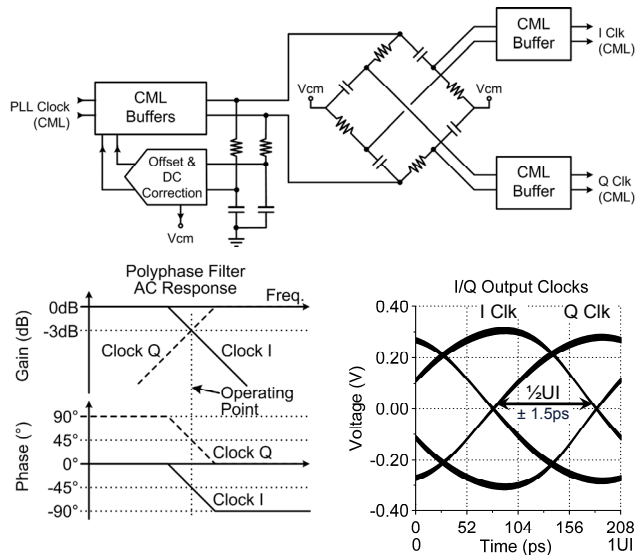


Figure 18.6.3: Polyphase IQ generator.

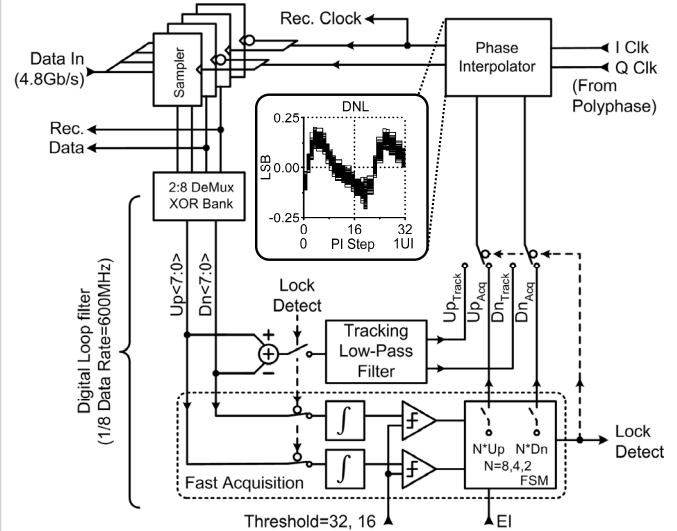


Figure 18.6.4: CDR with fast acquisition.

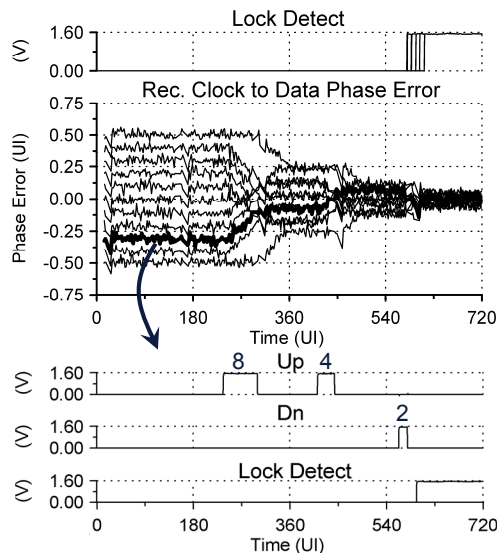


Figure 18.6.5: The CDR initial phase convergence.

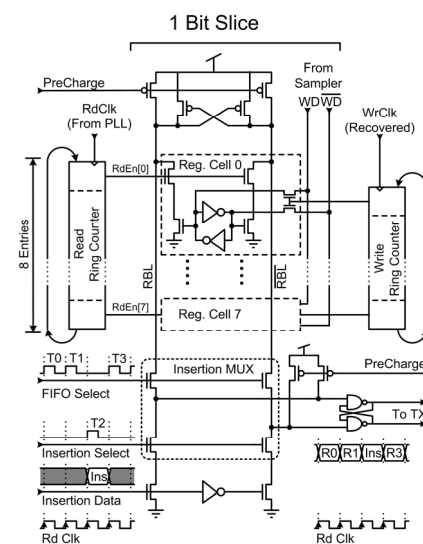


Figure 18.6.6: Retiming FIFO with integrated insertion MUX.

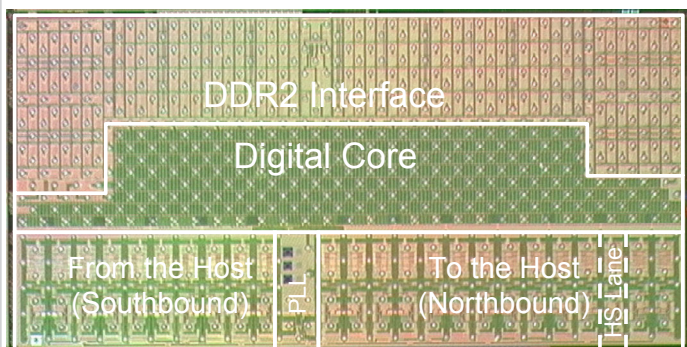


Figure 18.6.7: AMB die micrograph.